DOI:
10.1007/s11222-016-9633-6

Statistics & Computing manuscript No.
(will be inserted by the editor)

A multi-objective coordinate-exchange two-phase local search algorithm for multi-stratum experiments

Matteo Borrotti · Francesco Sambo · Kalliopi Mylona · Steven Gilmour

Received: date / Accepted: date

Abstract A multi-stratum design is a useful tool for industrial experimentation, where factors that have levels which are harder to set than others, due to time or cost constraints, are frequently included. The number of different levels of hardness to set defines the numbers of strata that should be used. The simplest case is the split-plot design, which includes two strata and two sets of factors defined by their level of hardness-to-set. In this paper, we propose a novel computational algorithm which can be used to construct optimal multi-stratum designs for any number of strata and up to six optimality criteria simultaneously. Our algorithm allows the study of the entire Pareto front of the optimization problem and the selection of the designs representing the desired trade-off between the competing objectives. We apply our algorithm to several real case scenarios and we show that the efficiencies of the designs obtained present experimenters with several good options according to their objectives.

M. Borrotti
Institute of Applied Mathematics and Information Technology, National Research Council of Italy, Milan, Italy
E-mail: matteo.borrotti@mi.imati.cnr.it

F. Sambo
Department of Engineering Information, University of Padua, Padua, Italy
E-mail: francesco.sambo@dei.unipd.it

K. Mylona
Department of Statistics, Universidad Carlos III de Madrid, Madrid, Spain
Statistical Sciences Research Institute, University of Southampton, Southampton, UK
E-mail: k.mylona@soton.ac.uk

S. Gilmour
Department of Mathematics, King's College London, Strand, London WC2R 2LS, UK
E-mail: steven.gilmour@kcl.ac.uk

Keywords Discrete optimization · Pareto-optimality · Split-plot design

1 Introduction

Industrial and laboratory experiments usually include factors with levels that are harder to set than others, whether because of the costs associated with setting the levels of the factors or because of the required time to set these levels in each run. It is clear that the best way to deal with such situations is to take into account the hard-to-set factors in a structured way, when designing the experiment, by ensuring that their levels do not have to be re-set in each subsequent run. This leads to a situation known as *restricted randomization*.

Generally, each level of hardness-to-set in factors which is taken into account in the design defines a stratum, as does each level of blocking, which restricts the randomization to allow for differences in the experimental conditions. Inside each stratum the factors are randomized keeping the other strata unchanged.

These multiple randomizations lead to multi-stratum designs: when there is only one restricted randomization (two strata) it is called a split-plot design, when there are two restricted randomizations it is called a split-split-plot design and so on. In the more general case where the strata are created, because of blocking structures and not because of the existence of hard-to-set factors, the designs are called randomized incomplete block designs. In this paper, following Trinca and Gilmour (2001), we refer to designs with factors in at least two strata as multi-stratum designs. However the methods presented are very general and, as will also be shown, can also be applied to completely randomized and randomized block design structures.

In Section 4, we present several real experiments in which multi-stratum designs were the best choice and yet the objectives were more complex than can be addressed by a single standard optimality criterion. In Trinca and Gilmour (1999), the main objective of the experiment was to discover how some factors that are involved in an extrusion process for mixing dough could be varied to control the properties of the pastry. The chosen design was a randomized incomplete block design that involved three factors and seven blocks with four observations each. The proposed analysis would involve fitting models to multiple responses, perhaps model selection and prediction for each, as well as interpretation of the best fitting models. These multiple analyses are not adequately addressed by a single optimality criterion.

In Ferryanto and Tollefson (2010), a split-split-plot design was used in a process the goal of which was to construct a container for contact lenses that must maintain its integrity for a long time and yet be easily opened. The peel behavior was considered as the design response and peel force and peel type were evaluated. The very-hard-to-set factor corresponded to the temperature and it was changed three times. The pressure levels form the hard-to-set factor and were changed nine times. Finally, within a particular pressure level, the dwell times were set in completely randomized order, forming two restricted randomizations. Again, meeting the practical objectives required multiple types of data analysis and a design which simply optimizes parameter estimation or prediction would not fully meet these objectives.

The general form of the model, derived from the randomization, for an experiment with N runs and s strata, with stratum i having n_i units within each unit at stratum $(i-1)$ and stratum 0 being defined as the entire experiment $(n_0 = 1)$ is given by

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \sum_{i=1}^{s} \mathbf{Z_i}\boldsymbol{\varepsilon_i}, \tag{1.1}$$

where $\mathbf{y}$ is an N-dimensional vector of responses ($N = \prod_{j=1}^{s} n_j$), $\mathbf{X}$ is an $N \times p$ model matrix, $\boldsymbol{\beta}$ is a p-dimensional vector containing the p fixed model parameters and $\mathbf{Z_i}$ is an $N \times b_i$ indicator matrix of zeros and ones for the units in stratum i (*i.e.* the $(k,l)^{th}$ element of $\mathbf{Z_i}$ is one if the k^{th} run belongs to the l^{th} block in stratum i and zero otherwise), $b_i = \prod_{j=1}^{i} n_j$. The vector $\boldsymbol{\varepsilon_i} \sim N(0, \sigma_i^2 \mathbf{I_{b_i}})$ is b_i-dimensional and contains the random effects and all random effects are uncorrelated. The main aim is usually to estimate the treatment parameters $\boldsymbol{\beta}$ but, in order to estimate their standard errors, it is also necessary to estimate the variance components σ_i^2, $i = 1, \ldots, s$.

Recently, computerized design search algorithms have gained popularity for constructing multi-stratum designs, especially split-plot and split-split-plot designs for different reasons. In fact, the lack of theoretical results, the small number of runs and/or whole plots in split-plot response surface experiments and, often, categorical factors in addition to quantitative factors can limit the applicability of classical techniques.

Possible solutions are approaches that are flexible with respect to the number of runs and number of whole plots. In Trinca and Gilmour (2001), a sequential method for constructing multi-stratum designs, from stratum to stratum and starting from the highest stratum, is presented, and an enhanced version of it is presented in Trinca and Gilmour (2015). The point-exchange algorithms for constructing D-optimal designs for split-plot response surface experiments proposed by Goos and Vandebroek (2001, 2003), and the coordinate-exchange algorithm for split-plot and split-split-plot experiments described in Jones and Goos (2007, 2009) are examples of such approaches.

A weakness of the aforementioned algorithms is that they focus entirely on optimizing a single criterion. In Lu et al. (2011), the Pareto Aggregating Point Exchange (PAPE) algorithm is proposed, to more efficiently explore candidate designs by considering the Pareto front, *i.e.* the set of different trade-offs between multiple optimality criteria. In that paper, the Pareto front approach for simultaneously considering multiple responses is adapted to design of experiments. In Sambo et al. (2014), the Pareto approach is applied to the construction of split-plot designs: the newly introduced Coordinate Exchange - Two Phase Local Search (CE-TPLS) algorithm extends the Jones and Goos (2007, 2012) coordinate-exchange (CE) algorithm with a two-phase local search approach (Paquete and Stützle, 2007). The output of the CE-TPLS algorithm is a set of non-dominated designs, *i.e.* a set in which no design is better than any of the others according to two optimality criteria (D-, I- criteria).

The most commonly used optimality criterion in the literature is D-optimality, which maximizes the determinant of the information matrix. In Jones and Goos (2012), the I-optimality criterion is used, which minimizes the average prediction variance, for generating split-plot response surface designs. The I-optimality criterion is usually more suitable when the goal is to make predictions. In Trinca and Gilmour (2015), in addition to D-optimality, A-optimality is also considered. This criterion results in minimizing the average variance of the estimates of the parameters.

The D-, I- and A-optimality criteria all assume that the estimation of the intercept is important and the I-

optimality criterion in particular is strongly driven by how well the intercept is estimated. It is unusual in practice that the intercept is a main concern and theoretically estimation of the intercept is meaningful only if the runs of the experiment can be considered to be a random sample of the population of runs of interest. Hence, additionally, the D_s, A_s- and I_D-criteria are considered, in each case excluding the intercept from the set of parameters whose estimation is to be optimized.

The purpose of this paper is to introduce an innovative multi-objective algorithm, to generalize the application of the CE-TPLS algorithm to: 1) search for any type of nested multi-stratum experiment, rather than just split-plots; and 2) optimize according to any combination of multiple optimality criteria (D-, A-, I-, D_s-, A_s- and I_D-criteria), rather than just I- and D-optimality.

Multi-objective optimization with respect to more than two criteria simultaneously is a very challenging problem and it has never been considered, to the best of our knowledge, in the existing literature on restricted randomized designs.

The remainder of the paper is organized as follows. In Section 2, we introduce the optimality criteria that will be considered. Section 3 presents the extended CE-TPLS algorithm, the multi-stratum two-phase local search (MS-TPLS) algorithm. Section 4 introduces four case studies and Section 5 compares the experimental results of our algorithm with both the results of the original coordinate-exchange algorithm and the results reported in the literature. Finally, Section 6 draws some conclusions and proposes directions for future research.

2 Optimal Multi-Stratum Designs

Given the vector of responses $\mathbf{y}$, the model matrix $\mathbf{X}$ and the $\mathbf{Z_i}$ indicator matrix for the units in each stratum i, the best linear unbiased estimator for the parameter vector $\boldsymbol{\beta}$ is the generalized least squares estimator

$$\hat{\boldsymbol{\beta}}_{GLS} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{y}.$$

This estimator has covariance matrix

$$Var(\hat{\boldsymbol{\beta}}_{GLS}) = \sigma^2(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1},$$

where $\mathbf{V} = \sum_{i=1}^{s} \eta_i \mathbf{Z_i}'\mathbf{Z_i}$, $\eta_i = \sigma_i^2/\sigma^2$ and $\sigma^2 = \sigma_s^2$.

In practice, the variance components have to be estimated and this is usually done through residual maximum likelihood (REML), as recommended in Letsinger et al. (1996) and Gilmour and Trinca (2000).

As described by Goos (2006) and Jones and Nachtsheim (2009), there are several approaches for setting up multi-stratum response surface designs which have gained popularity in the literature. Although it has been common in the optimal design literature to consider just one optimality criterion for the construction of the optimal design, depending on the goal of the experiment (see for instance Trinca and Gilmour (2001), Goos and Vandebroek (2003), Goos and Donev (2006), Jones and Goos (2012)), a multi-objective approach provides more flexibility to the experimenter. We focus on the D-, A- and I-optimality criteria and then on the D_s-, A_s- and I_D-optimality criteria. In each case, the aim is to optimize a scalar-valued function of the design matrix for given point prior estimates of the ratios of variance components.

2.1 D-optimality criterion

The most commonly used optimality criterion for selecting experimental designs is the D-optimality criterion. This criterion seeks to minimize the generalized variance of the parameter estimates, which is done by minimizing the determinant of the variance-covariance matrix of the factor effects' estimates or, equivalently, by maximizing the determinant of the information matrix about $\hat{\boldsymbol{\beta}}$. For a multi-stratum design, the information matrix is given by

$$\mathbf{M} = \mathbf{X}'\mathbf{V}^{-1}\mathbf{X}, \tag{2.1}$$

when the GLS estimator is used. As discussed in Goos (2002) and Trinca and Gilmour (2015), usually some point prior estimate of the ratios of variance components $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_s)$ is used, as the optimal designs depend on the relative magnitude of σ_i^2 and σ^2, $\boldsymbol{\eta}$, but not on their absolute magnitude.

For simplicity and uniformity with the other criteria, we define the objective function for D-optimality as

$$f_D(\mathbf{d};\boldsymbol{\eta}) = \left(\frac{1}{\det(\mathbf{M})}\right)^{1/p},$$

where $\mathbf{d}$ is the design with information matrix $\mathbf{M}$ and p is the number of model parameters. The $f_D(\mathbf{d};\boldsymbol{\eta})$ function has to be minimized.

2.2 A-optimality criterion

Like the D-optimality criterion, the A-optimality criterion is a general measure of the size of the variance-covariance matrix $\mathbf{M^{-1}}$. A-optimality is based on the sum of the variances of the estimated parameters for the model, which is the same as the sum of the diagonal elements, or trace, of $\mathbf{M^{-1}}$. This criterion results in minimizing the average variance of the estimates of the regression coefficients. We define the objective function for A-optimality as

$$f_A(\mathbf{d};\boldsymbol{\eta}) = trace(\mathbf{M^{-1}}).$$

The objective function for A-optimality, $f_A(\mathbf{d};\boldsymbol{\eta})$, has to be minimized.

2.3 I- and I_D-optimality criteria

An I-optimal split-plot design minimizes the average prediction variance

$$f_I(\mathbf{d};\boldsymbol{\eta}) = \frac{\int_\chi \mathbf{f}'(\mathbf{x})(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{f}(\mathbf{x})d\mathbf{x}}{\int_\chi d\mathbf{x}},$$

where χ represents the design region. When there are k treatment factors, i.e. explanatory variables whose levels are controlled by the experimenter, and the experimental region is $[-1,+1]^k$, this expression can be rewritten as (Jones and Goos, 2012)

$$f_I(\mathbf{d};\boldsymbol{\eta}) = \text{tr}\left[(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{B}\right], \tag{2.2}$$

where $\mathbf{B}$ is the moments matrix,

$$\mathbf{B} = 2^{-k}\int_\chi \mathbf{f}(\mathbf{x})\mathbf{f}'(\mathbf{x})d\mathbf{x}.$$

An expression for calculating the moments matrix for a cuboidal design region is given by Hardin and Sloane (1991). The matrix $\mathbf{B}$ has a very specific structure for a full quadratic model,

$$\mathbf{B} = \begin{bmatrix} 1 & \mathbf{0}'_{\mathbf{k}} & \mathbf{0}'_{\mathbf{k}^*} & \frac{1}{3}\mathbf{1}'_{\mathbf{k}} \\ \mathbf{0}_{\mathbf{k}} & \frac{1}{3}\mathbf{I}_{\mathbf{k}} & \mathbf{0}_{\mathbf{k}\times\mathbf{k}^*} & \mathbf{0}_{\mathbf{k}\times\mathbf{k}} \\ \mathbf{0}_{\mathbf{k}^*} & \mathbf{0}_{\mathbf{k}^*\times\mathbf{k}} & \frac{1}{9}\mathbf{I}_{\mathbf{k}^*} & \mathbf{0}_{\mathbf{k}^*\times\mathbf{k}} \\ \frac{1}{3}\mathbf{1}_{\mathbf{k}} & \mathbf{0}_{\mathbf{k}\times\mathbf{k}} & \mathbf{0}_{\mathbf{k}\times\mathbf{k}^*} & \frac{1}{45}(4\mathbf{I}_{\mathbf{k}}+5\mathbf{J}_{\mathbf{k}}) \end{bmatrix},$$

where $k^* = k(k-1)/2$ is the number of two-factor interaction effects. When a sub-model is considered the corresponding elements of $\mathbf{B}$ should be eliminated, e.g. if we consider the sub-model that does not include the quadratic terms we need to delete the last set of columns and the last set of rows of the matrix. For the I_D-criterion, see Trinca and Gilmour (2015), the first row and column are deleted. The objective functions for I- and I_D-optimality have to be minimized and will be named $f_I(\mathbf{d};\boldsymbol{\eta})$ and $f_{I_D}(\mathbf{d};\boldsymbol{\eta})$, respectively.

2.4 D_s and A_s-optimality criteria

Let $\boldsymbol{\beta}_i$ be the model parameter vector ($p_i - 1$ parameters, excluding the intercept) to be estimated in stratum i. Let $\mathbf{X_i}$ be the $m_i \times (p_i-1)$ associated model matrix where m_i is the number of units in this stratum. The partition of interest of the variance covariance matrix of $\widehat{\boldsymbol{\beta}_i}$ is $(\mathbf{M}_i^{-1})_{22} = [\mathbf{X_i}'(\mathbf{I}-\frac{1}{m_i}\mathbf{1}\mathbf{1}')\mathbf{X_i}]^{-1}$. Thus for D_s, we minimize $f_{D_s}(\mathbf{d};\boldsymbol{\eta}) = |(\mathbf{M}_i^{-1})_{22}|$ and for A_s-optimality we minimize $f_{A_s}(\mathbf{d};\boldsymbol{\eta}) = trace(\mathbf{W_i}(\mathbf{M}_i^{-1})_{22})$, where $\mathbf{W_i}$ is a diagonal matrix of weights, with the weights scaled so that $trace(\mathbf{W_i}) = 1$, which allows the estimation of some parameters to be given more weight than that of others. In the experiments reported in the Results section, we use a weight matrix for A_S such that the relative weights are 1/4 for each quadratic effect, 1/2 for each interaction term and 1 for the main effects.

Note that independently of the criterion all the optimal designs depend on the variance ratio $\boldsymbol{\eta}$ through the covariance matrix $\mathbf{V}$.

3 The coordinate-exchange two-phase local search algorithm

When multiple criteria are concurrently targeted in optimal design of experiments, the result of the optimization is in general no longer a single optimal design, but rather a set of designs, representing several trade-offs between the competing objectives (Paquete and Stützle, 2007).

More precisely, for the multi-objective optimization problem of minimizing n optimality criteria, candidate designs are evaluated according to an objective function vector $\overline{f} = (f_{c_1}, f_{c_2}, \ldots, f_{c_n})$, where $c_1 \ldots c_n$ are the different criteria. Given two designs $\mathbf{d}$ and $\mathbf{d}'$, we say that $\mathbf{d}$ *dominates* $\mathbf{d}'$ ($\mathbf{d} \prec \mathbf{d}'$) iff $\overline{f}(\mathbf{d}) \neq \overline{f}(\mathbf{d}')$ and $f_c(\mathbf{d}) \leq f_c(\mathbf{d}')$, $\forall c \in \{c_1 \ldots c_n\}$.

If no $\mathbf{d}'$ exists such that $\mathbf{d}' \prec \mathbf{d}$, the design $\mathbf{d}$ is called *Pareto-optimal*. In this context, the goal of multi-objective optimal design is to determine (or approximate) the set of all Pareto-optimal designs, whose image in the multi-objective space is called the *Pareto front*.

To tackle the multi-objective optimization problem of optimal multi-stratum experiment design we exploit the *two-phase local search* (TPLS) approach (Dubois-Lacoste et al., 2011). TPLS is a general algorithmic framework for multi-objective optimization composed, as the name suggests, of two phases. In the first phase, a single-objective local search algorithm generates a high-quality design for each of the n objectives. These solutions serve as starting points of the second phase, in which the local search algorithm is exploited to find a sequence of locally optimal designs: each design is obtained starting the search from one of the previous local optima and optimizing a different *scalarization*, *i.e.* a different weighted sum of the n objective functions into a single scalar function.

The algorithm we propose, multi-stratum two-phase local search (MS-TPLS), stems from the coordinate-exchange two-phase local search (CE-TPLS) algorithm for the D- and I-optimal design of split-plot experiments presented in Sambo et al. (2014), but is the result of two major extensions: i) the concurrent optimization of any combination of six design optimization criteria, rather than just the D and I criteria; and ii) the application to any type of multi-stratum designs, rather than just split-plot designs. From an algorithmic point of view, the first major extension is obtained in line 6 in MS-Opt (Fig.1). More precisely, the function Scores contains the implementation of all the considered criteria. The second major extension is codified from line 10 to line 20 of MS-Opt.

The single-objective local search component of the (MS-TPLS) algorithm minimizes a scalarization of the objective functions for all criteria in the form

$$f_W(\mathbf{d};\boldsymbol{\eta}) = \sum_{c\in C} \alpha_c f_c(\mathbf{d};\boldsymbol{\eta}) = \overline{\alpha}\cdot\overline{f}, \qquad \sum_{c\in C}\alpha_c = 1 \quad (3.1)$$

where C is the set of criteria to be minimized, f_c is the objective function for criterion c and the parameters $\overline{\alpha}$ control the relative weight of each objective function. For the scalarization to be unbiased, all objective functions should lie in the range [0,1]: before computing each scalarization, thus, we dynamically normalize each objective function value as

$$f_c^{norm}(\mathbf{d};\boldsymbol{\eta}) = \frac{f_c(\mathbf{d};\boldsymbol{\eta}) - f_c^{min}(\mathbf{d};\boldsymbol{\eta})}{f_c^{max}(\mathbf{d};\boldsymbol{\eta}) - f_c^{min}(\mathbf{d};\boldsymbol{\eta})},$$

where $f_c^{min}(\mathbf{d};\boldsymbol{\eta})$ and $f_c^{max}(\mathbf{d};\boldsymbol{\eta})$ are the minimum and maximum values of the objective function, among all the designs encountered by the algorithm from the beginning of the run.

The pseudocode of the single-objective local search procedure, MS-Opt, is given in Figure 1. MS-Opt is designed to either start from a given initial design, passed as the input *initDesign*, or to generate one at random with the SampleDesign procedure (Fig. 2). The initial design is then improved by iteratively replacing values for each factor in each unit of each stratum, until no further local change can increase the weighted sum $f_W(\mathbf{d};\boldsymbol{\eta})$ of the objective functions. Objective functions for the specified criteria are computed from a design matrix, as explained in Section 2, with the function Scores and returned as a vector. The procedure is repeated for a given number of iterations and the best design is returned.

To obtain a good approximation of the Pareto front in the space of optimal multi-stratum designs, we exploited the MS-opt algorithm in a TPLS framework, whose pseudocode is reported in Figure 3 (multi-stratum two-phase local search algorithm, MS-TPLS).

The algorithm exploits the *archive* data structure to store all the designs generated during the search and uses them to compute, at each iteration, the set of non-dominated designs forming the Pareto front.

As indicated in the pseudocode, the algorithm first computes n high-quality solutions, one for each optimization criterion, by calling the MS-Opt algorithm, with $\overline{\alpha}$ set to 1 for the desired criterion and 0 otherwise and with the number of iterations set to $initIterations$.

Next, the algorithm generates a sequence of scalarizations by iteratively sampling an initial design *initDesign* from the Pareto Front, sampling uniformly at random the scalarization weights $\overline{\alpha}$ and running one iteration of MS-Opt. Previous studies (Sambo et al., 2014) demonstrated that sampling at random appears to achieve the best balance between diversification (i.e. exploration) of the solutions and intensification (i.e. exploitation) of the search on promising regions of the multi-objective space. Finally, the algorithm removes dominated designs from the *archive* and returns it.

4 Problem Instances

In this section, we present five problem instances chosen from the literature. Each of the five instances represents a different experimental scenario and differs from the others in terms of the number of factors, strata and available runs.

4.1 *Instance 1*: The cassava bread experiment

In the first experiment, which we use to illustrate the optimization algorithm in the simplest possible context, a completely randomized design was used. The example is based on work presented by Escouto (2000), which performed an experiment in order to find a recipe for gluten-free bread based on cassava flour. Twenty-six observations were taken and a second-order model was considered. The experiment involved three factors at three levels: x_1, the amount of powdered albumen (egg white); x_2, the amount of yeast; x_3, the amount of ground cassava flour. Several characteristics of the bread were evaluated as response variables (i.e. crust color, break symmetry, crust characteristics, crumb color,

```
MS-Opt(criteria, ᾱ, iterations, initDesign)
bestScore = +INF
for it = 1 to iterations
    if initDesign == NULL
        curDesign = SampleDesign()
    else curDesign = initDesign
    curScore = ᾱ · Scores(curDesign, criteria)
    improvement = TRUE
    while improvement
        improvement = FALSE
        for each stratum s
            for each unit u of s
                for each factor f in s
                    for each of the other available values of f
                        nextDesign = curDesign
                        Assign the value to all elements of matrix tmpDesign
                        at column f and at the rows corresponding to u
                        nextScore = ᾱ · Scores(nextDesign, criteria)
                        if nextScore < curScore
                            curDesign = nextDesign
                            curScore = nextScore
                            improvement = TRUE
    if curScore < bestScore
        bestDesign = curDesign
        bestScore = curScore
return bestDesign
```

Fig. 1: Pseudocode of sub-routine MS-Opt.

```
SampleDesign()
for each stratum s
    for each unit u of s
        for each factor f in s
            Sample one of the available values of f at random
            Assign the value to all elements of matrix design
            at column f and at the rows corresponding to u
return design
```

Fig. 2: Pseudocode of sub-routine SampleDesign.

structure of cells of the crumb, crumb texture, flavor and taste) in order to obtain a formulation which presents features similar to the ones of wheat-based white bread. In Gilmour and Trinca (2012), several alternative designs are presented. Among others, we considered the one constructed with D_s and A_s-criteria, which gave identical designs.

4.2 *Instance 2*: The pastry dough experiment

In the second experiment, a randomized incomplete block design was used. The experiment was described in Trinca and Gilmour (1999) and the main objective was to discover how some factors that are involved in an extrusion process for mixing dough could be varied to control the properties of the pastry. It involved seven blocks, three factors and twenty-eight runs. Nine responses were mea-

```
MS-TPLS(criteria, initIterations, scalarizations)
archive = ∅
n = LENGTH(criteria)
for c = 1 to n
    ᾱ = vector of n zeros, with element c = 1
    design = MS-Opt(criteria, ᾱ, initIterations, NULL)
    Add design to archive
for sc = 1 to scalarizations
    Sample initDesign from the Pareto Front of archive
    Sample ᾱ at random
    design = MS-Opt(criteria, ᾱ, 1, initDesingg)
    Add design to archive
Remove dominated designs from archive
return archive
```

Fig. 3: Pseudocode of the multi-stratum two-phase local search algorithm (MS-TPLS)

sured: three measuring the size of the pastry, three measuring the strength of the pastry and three measuring the colour of the pastry. The three factors, each with three levels, were: x_1, flow rate; x_2, moisture content; x_3, screw speed. D_s and A_s-optimal designs were constructed for this scenario. The two criteria gave identical designs under the second-order model.

4.3 *Instance 3*: The protein extraction experiment

In the third experiment, a split-plot design was used. In Trinca and Gilmour (2001), an experiment is described that investigates the effect of five factors on protein extraction. More precisely, a mixture containing two valuable proteins, among other components, is considered after fermentation and purification processes. The experiment was intended to separate the two proteins from the mixture, and the responses were the yields and purities of the two proteins. The factors were: x_1, the feed position for the inflow of a mixture, which is hard to set; x_2 the feed flow rate; x_3 the gas flow rate; x_4 the concentration of the first protein; x_5, the concentration of the second protein. Three levels were used for each factor. The split-plot design was set up as follows: one whole-plot factor, four subplot factors, and twenty-one whole plots of size two. This scenario was used by Jones and Goos (2012) in order to compare the D-optimal and the I-optimal designs under a second-order response surface model.

4.4 *Instance 4*: Contact lens package foil lidding experiment

The fourth experiment concerns the study of foil lidding for sealing packages of contact lenses, which are available in many different structures and compositions. The main feature of this container for contact lenses is that it must maintain its integrity for a long time and yet be easily opened. For this experiment, Ferryanto and Tollefson (2010) proposed to use a split-split plot design and they considered a model which included up to three-factor interactions. Three factors were considered: x_1 set temperature, x_2 seal pressure, and x_3 dwell time, each with three levels. The peel behavior was considered as the response and peel force and peel type were evaluated. The whole plots correspond to the temperature and it was changed three times. The pressure levels form three subplots and were changed nine times. Finally, within a particular pressure level, the dwell times were set in a completely randomized order, forming three subsubplots. The total number of runs was set to twenty-seven and the design was replicated twice.

4.5 *Instance 5*: Baja car prototyping experiment

The fifth experiment used a split-split-split plot design. The experiment is based on the work of Lee Ho et al. (2012), who proposed an experimental design for assembling a Baja car prototype. The application involving the Baja car depends on nine factors: x_1 type of ceasefire plate, x_2 driven pulley cam angle, x_3 driven pulley material, x_4 driven pulley spring type, x_5 driven pulley spring pressure, x_6 material of driver pulley cap, x_7

driver pulley mass, x_8 driver pulley spring type x_9 tire pressure. Each of the factors has two levels. The prototyping experiment is based on four strata, organised as follows: two changes in the first stratum (size 16), eight changes in the second stratum (size 4), sixteen changes in the third stratum (size 2) and thirty-two changes in the fourth stratum. The total number of runs of the experiment is equal to the number of treatment changes in the last stratum. In this case, we consider a model with main effects. The objective of the experiment was to maximize the performance of the vehicle on two tests: (i) acceleration test, which evaluates the time that the vehicle takes to cover a distance of 30 meters starting from a complete stop, and (ii) velocity test, which measures the final velocity reached by the vehicle at the 100 meters mark.

5 Results and Comparison

In this section we investigate the performance of our algorithm in the five instances. First of all, the free parameters of our algorithm have been set in accordance with the results obtained in Sambo et al. (2014), in which the best balance between diversification of the solutions and intensification of the search has been obtained by sampling initial designs from the Pareto front and sampling the α weight at random from $[0, 1]$. The value of η_i, $i = 1, \ldots, s$ with s the number of strata in the experiment, is 1 for all the reported results. Furthermore, the results were obtained with $n \times 10$ restarts, where n is the number of criteria simultaneously optimized, of the MS-TPLS algorithm, each composed of $n \times 2$ initial single-objective iterations followed by $100 - (n \times 2)$ scalarizations, for a total of $n \times 1000$ calls to the MS-Opt function. In Sambo et al. (2014), the authors have empirically demonstrated that 1000 scalarizations are sufficient to achieve a good approximation of the Pareto front. More precisely, if we want to optimize 3 criteria (i.e. I, D, A) then we have 30 restarts each composed of 2 + 2 + 2 initial single-objective iterations followed by 94 scalarizations.

We have grouped together the criteria which allow for estimation of the intercept, namely A-, I- and D-optimality to try to find designs which are good for all of these criteria. Separately, we study the D_s, I_D and A_s criteria.

Figures 4a, 5a, 6a, 7a and 8a show the Pareto front obtained by taking the set of non-dominated designs from the results of 30 random restarts of the MS-TPLS algorithm for the five instances, concurrently optimizing the criteria I, D and A. Along the x-axis is reported the value of the I-optimality criterion and along the y-axis the value of the A-optimality criterion. The third dimension, represented by the D-optimality criterion, is coded as shades of gray, the lower the darker. Figures 4b, 5b, 6b, 7b and 8b show the same plots obtained for the criteria I_D, D_s and A_s.

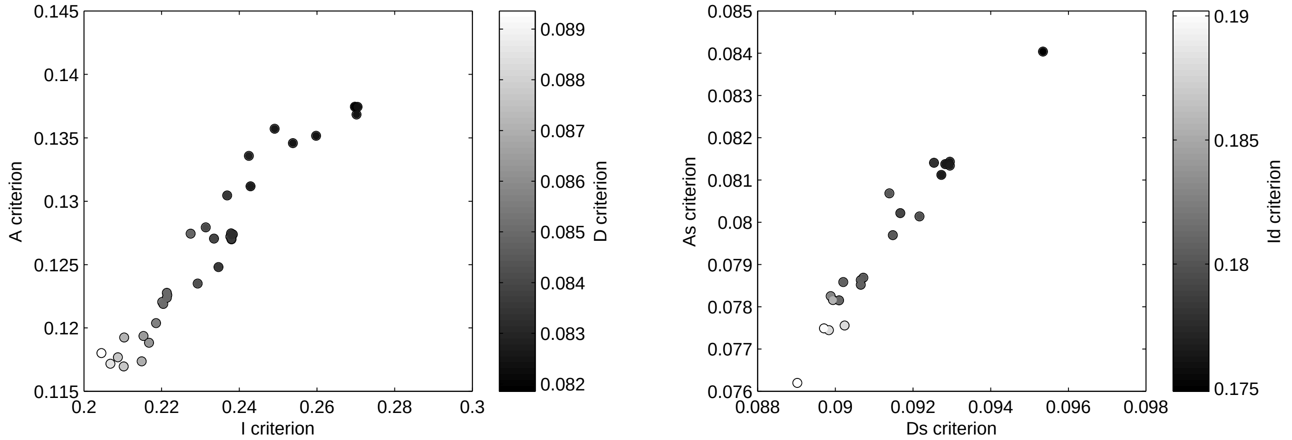
As is clear from the figures, the search space characteristics and the shape of the Pareto front strongly depend on the model type and to a lesser extent on the multi-stratum type of the design. For the full quadratic model (Figures 4, 5 and 6), one can observe a strong but imperfect correlation between the A and I criteria and a negative correlation of the two with the D-criterion of the solutions in the Pareto front. The same pattern applies to the A_s and D_s criteria, which are both strongly correlated with each other and negatively correlated with the I_D criterion. In each figure, the criteria represented along the horizontal and vertical axes of the plots were chosen in order to illustrate the notable correlations.

These correlations are not completely surprising, since they are in accordance with results from the literature that demonstrate the conflict between D- and I-optimality (Hardin and Sloane (1993); Jones and Goos (2012)). In addition, as can be seen in (2.2), the I and I_D criteria can be re-expressed as trace criteria, just like the A and A_s criteria. The Pareto front exhibits a lot of non-dominated solutions, reflecting the complexity of the model and its sensitivity to small changes in the design matrix. For example, all 25 design matrices, and the related values of the I_D, D_s and A_s-criterion functions, of the Pareto front in Instance 2 (Figure 5b) are provided in the supplementary material.

The situation is completely different for the model with main effects and interactions (Figure 7), where the Pareto fronts are much simpler and the patterns of correlation between criteria change. When the model is reduced to just the main effects (Figure 8), the Pareto front for both triplets of criteria collapses to just one point.

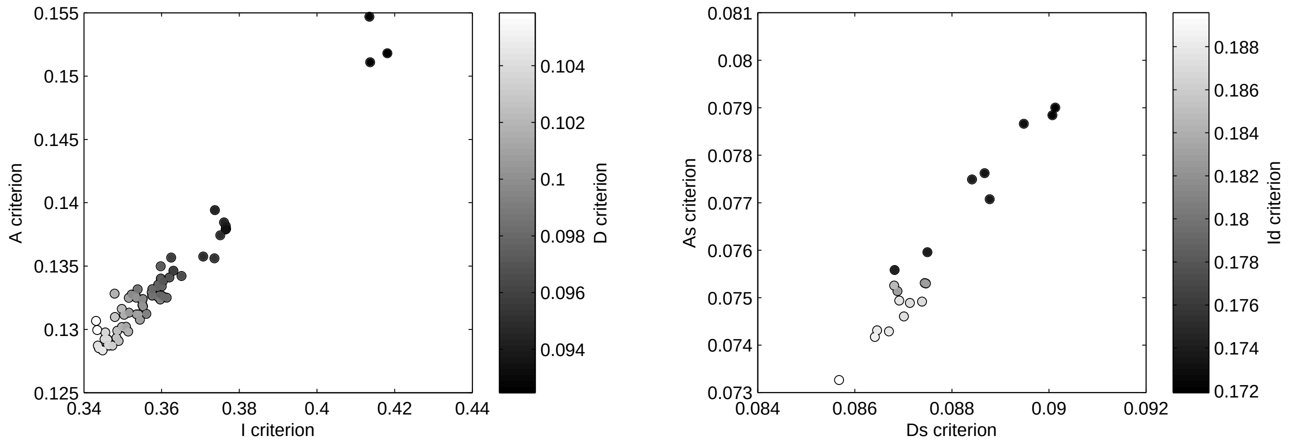
The best compromise between the I-, D- and A-criteria can be identified as the closest design to the *utopia* point, which is the ideal point in the three-objective space with the minimum value of the I-, D- and A-criteria (Lu et al., 2011). We name this design the *I,D,A-symmetrical* design (or *I_D,D_s,A_s-symmetrical* design when considering I_D-, D_s- and A_s-criteria) and select it for further inspection, together with the optimal designs according to each separate criterion.

Table 1 reports, for each instance, the I-, D- and A-criterion function values of the selected designs obtained by our algorithm, of the designs obtained by the original, single objective CE algorithm of Jones and Goos (2007) (run for 1000 restarts for each criterion) and, for completeness, of the designs reported in the



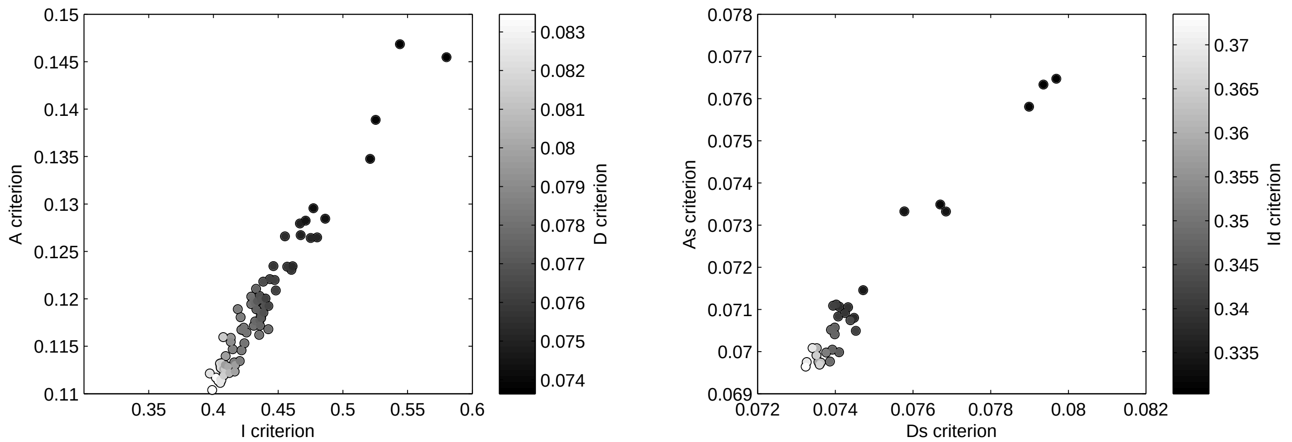
(a) Pareto front for the A-, I- and D- criteria. (b) Pareto front for the A_s-, D_s- and I_D- criteria.

Fig. 4: Pareto fronts for Instance 1: completely randomized design, quadratic model



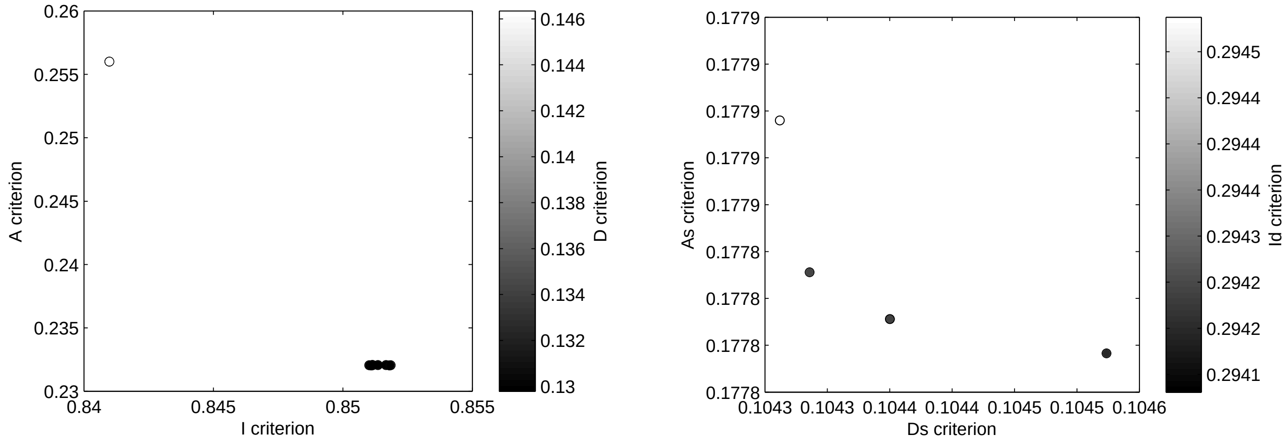
(a) Pareto front for the A-, I- and D- criteria. (b) Pareto front for the A_s-, D_s- and I_D- criteria.

Fig. 5: Pareto fronts for Instance 2: randomized incomplete block design, quadratic model



(a) Pareto front for the A-, I- and D- criteria. (b) Pareto front for the A_s-, D_s- and I_D- criteria.

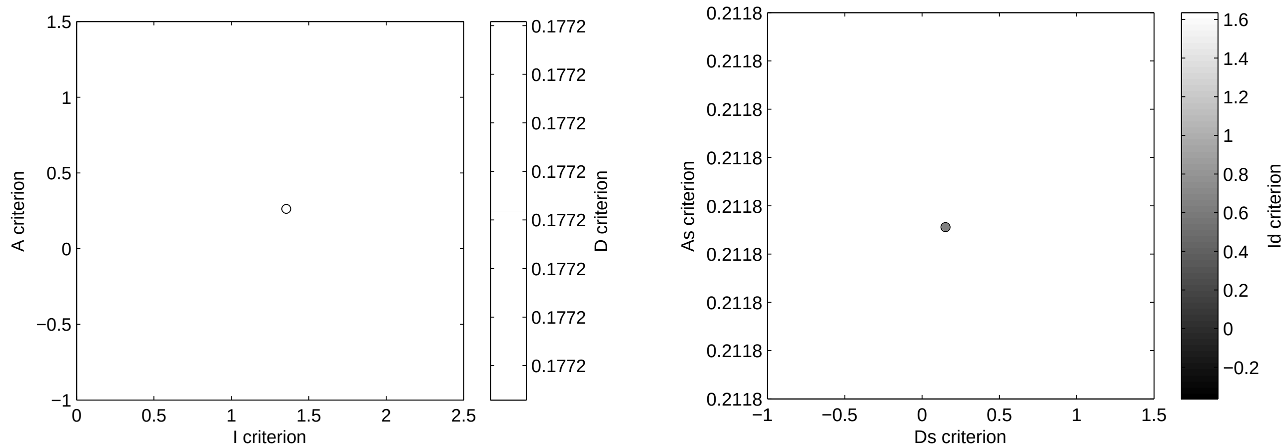
Fig. 6: Pareto fronts for Instance 3: split-plot design, quadratic model



(a) Pareto front for the A-, I- and D- criteria.

(b) Pareto front for the A_s-, D_s- and I_D- criteria.

Fig. 7: Pareto fronts for Instance 4: split-split plot design, main effects + interactions model



(a) Pareto front for the A-, I- and D- criteria.

(b) Pareto front for the A_s-, D_s- and I_D- criteria.

Fig. 8: Pareto fronts for Instance 5: split-split-split plot design, main effects model

original paper of each instance. Entries in bold correspond to the best values across each row. In a similar way, Table 2 reports the I_D-, D_s- and A_s-criteria of the selected designs obtained by our algorithm for each instance, of the designs computed with the CE algorithm and of the designs reported in the original papers.

By comparing the values in Table 1 and Table 2, one can see that the MS-TPLS algorithm is almost always able to find the best design according to all criteria, with the exception of few cases (the D-optimal design for Instance 2, the I- and D-optimal designs for Instance 3, and the I_D-optimal design for Instance 3). A possible explanation for the latter cases is that our algorithm for the construction of the Pareto front, using the same number of random starts as the CE algorithm, tries to optimize three criteria simultaneously, in contrast to the CE algorithm which concentrates in a single optimality criterion. For Instances 1 and 4, the best designs are found by both the MS-TPLS algorithm and the CE algorithm. The CE algorithm, though it is a single-objective optimization algorithm, does not perform so well as the MS-TPLS algorithm for the generation of the I-, D- and I_D-optimal design for Instance 2 and for the generation of the A-, D_s- and A_s-optimal design for Instance 3.

We note that the I-optimal design is also selected as the I,D,A-symmetrical design in Instance 3. The same is true for Instance 4, where the I_D,D_s,A_s-symmetrical design corresponds to the I_D-optimal design. In Instances 1 and 2, the selected optimal designs are the same for the D_s- and A_s-optimality criteria. Instance 5 leads to only two possible design matrices in the Pareto front, both of them with the same value of the considered optimality criteria. The same designs are also

Table 1: I-, D- and A-criteria of the I-optimal, D-optimal, A-optimal and I,D,A-symmetrical designs identified by the MS-TPLS algorithm (columns 1-4), of the I-optimal, D-optimal and A-optimal designs identified by the CE algorithm (columns 5-7) and of the design reported in the original paper (column 8) for each of the five instances, with $\eta = 1$. Entries in bold correspond to the best values across each row.

		MS-TPLS				CE			Original paper	
		I,D,A-sym	I-opt	D-opt	A-opt	I-opt	D-opt	A-opt		
Instance 1	I	0.2068	**0.2045**	0.2698	0.2103	**0.2045**	0.2698	0.2103	0.2698	
	D	0.0884	0.0894	**0.0819**	0.0876	0.0894	**0.0819**	0.0876	**0.0819**	
	A	0.1172	0.1180	0.1375	**0.1170**	0.1180	0.1375	**0.1170**	0.1375	
Instance 2	I	0.3462	**0.3431**	0.4181	0.3448	0.3453	0.4174	0.3472	0.4174	
	D	0.1031	0.1058	0.0925	0.1045	0.1075	**0.0922**	0.1042	**0.0922**	
	A	0.1288	0.1307	0.1518	**0.1283**	0.1321	0.1514	0.1296	0.1514	
Instance 3	I	0.3972	0.3972	0.5801	0.3991	0.3960	0.6552	0.4040	0.6551	**0.3940**
	D	0.0823	0.0823	0.0736	0.0835	0.0864	0.0734	0.0839	**0.0732**	0.0858
	A	0.1121	0.1121	0.1455	**0.1104**	0.1127	0.1598	0.1113	0.1620	0.1118
Instance 4	I	0.8510	**0.8410**	0.8514	0.8518	**0.8410**	0.8514	0.8518	0.8704	
	D	0.1299	0.1463	**0.1298**	0.1318	0.1463	**0.1298**	0.1301	0.1929	
	A	**0.2321**	0.2560	**0.2321**	**0.2321**	0.2560	**0.2321**	**0.2321**	0.2831	
Instance 5	I	**1.3542**	**1.3542**	**1.3542**	**1.3542**	**1.3542**	**1.3542**	**1.3542**	**1.3542**	
	D	**0.1772**	**0.1772**	**0.1772**	**0.1772**	**0.1772**	**0.1772**	**0.1772**	**0.1772**	
	A	**0.2625**	**0.2625**	**0.2625**	**0.2625**	**0.2625**	**0.2625**	**0.2625**	**0.2625**	

found by the CE algorithm. This is due to the simplicity of the model considered for this instance, the main effects model.

Table 3 reports the percentage efficiency gains by the I,D,A-symmetrical and I_D,D_s,A_s-symmetrical designs, with respect to the best design for each of the six criteria. The efficiency gain is calculated as

$$\%EFF_{gain}^{c,\text{c-opt}} = 100 \times \left(\frac{f_c^{\text{c-opt}}}{f_c^{sym}} - 1 \right), \tag{5.1}$$

where c identifies the criterion (I, D, A, I_D, D_s and A_s) for which efficiency is computed, $f_c^{\text{c-opt}}$ is the value of this criterion for the c-optimal design and f_c^{sym} is the value of the same criterion for the I,D,A-symmetrical or I_D, D_s, A_s-symmetrical design (the former if c is one of the I, D, A criteria and the latter if it is one of I_D, D_s, A_s criteria).

For example, considering Instance 1 we aim at calculating the efficiency gain of the I,D,A-symmetrical design, according to the I-, D- and A-criteria, with respect to the best D-optimal design. From Table 1 we select $f_I^{sym} = 0.2068$, $f_D^{sym} = 0.0884$ and the $f_A^{sym} = 0.1172$ and the corresponding values for the D-optimal design, $f_I^{\text{D-opt}} = 0.2698$, $f_D^{\text{D-opt}} = 0.0819$ and $f_A^{\text{D-opt}} = 0.1375$. We can see thus that, even though the I,D,A-symmetrical design has a loss of 7.35% in terms of D-efficiency, this is compensated by a gain of 30.46% in I-efficiency and 17.32% in A-efficiency.

By studying the table, one can see that the percentage of efficiency lost by the symmetrical design on the optimized criterion is almost always compensated by an equal or higher percentage gain on at least one other criterion, and often on both of the other criteria. In absolute terms, the percentage of efficiency loss on the optimized criterion is always lower than 12% for the I,D,A-symmetrical design and than 3.5% for the I_D, D_s, A_s-symmetrical design, while the gain on the other criteria can reach more than 60% in the first case and more than 11% in the second. Such results support the effectiveness of MS-TPLS in searching for the best compromises between different criteria.

Moreover, we calculated the variances of the parameter estimates for each design and for $\eta = 1, 10, 100$. The results are in accordance to the specific optimality of each design. Hence, the $D(D_s)$-optimal design is doing better with respect to the variances of the main effects and the two-factor interactions and the $I(I_D)$-optimal design is doing better with respect to the quadratic effects (in the cases where we have a second order model). The $A(A_s)$-optimal designs agree more with the $D(D_s)$ or the $I(I_D)$-optimal designs, depending on the case and in accordance to the imperfect correlations demonstrated in Figures 4-8. We have added the variances for Instance 3 and $\eta = 1, 100$ to the supplementary material.

Finally, in Table 4 we report the execution time for each instance and for different numbers of criteria in

Table 2: I_D-, D_s- and A_s-criteria of the I_D-optimal, D_s-optimal, A_s-optimal and I_D,D_s,A_s-symmetrical designs identified by the MS-TPLS algorithm (columns 1-4), of the I_D-optimal, D_s-optimal and A_s-optimal designs identified by the CE algorithm (columns 5-7) and of the design reported in the original paper (column 8) for each of the five instances, with $\eta = 1$. Entries in bold correspond to the best values across each row.

		MS-TPLS				CE			Original paper	
		I_D, D_s, A_s-sym	I_D-opt	D_s-opt	A_s-opt	I_D-opt	D_s-opt	A_s-opt		
Instance 1	I_D	0.1803	**0.1749**	0.1902	0.1902	**0.1749**	0.1902	0.1902	0.1902	
	D_s	0.0907	0.0953	**0.0890**	**0.0890**	0.0953	**0.0890**	**0.0890**	**0.0890**	
	A_s	0.0787	0.0840	**0.0762**	**0.0762**	0.0840	**0.0762**	**0.0762**	**0.0762**	
Instance 2	I_D	0.1742	**0.1719**	0.1896	0.1896	0.1724	0.1896	0.1908	0.1896	
	D_s	0.0868	0.0901	**0.0857**	**0.0857**	0.0909	**0.0857**	0.0860	**0.0857**	
	A_s	0.0756	0.0790	**0.0733**	**0.0733**	0.0797	**0.0733**	0.0738	**0.0733**	
Instance 3	I_D	0.3347	0.3303	0.3735	0.3735	**0.3283**	0.4346	0.3883	0.4148	0.3350
	D_s	0.0747	0.0797	**0.0732**	**0.0732**	0.0817	0.0736	0.0750	0.0733	0.0866
	A_s	0.0715	0.0765	**0.0696**	**0.0696**	0.0789	0.0723	0.0699	0.0725	0.0847
Instance 4	I_D	**0.2941**	**0.2941**	0.2945	0.2942	**0.2941**	0.2945	0.2942	0.3889	
	D_s	0.1044	0.1044	**0.1043**	0.1046	0.1044	**0.1043**	0.1046	0.1656	
	A_s	**0.1778**	**0.1778**	0.1779	**0.1778**	**0.1778**	0.1779	**0.1778**	0.2500	
Instance 5	I_D	**0.6354**	**0.6354**	**0.6354**	**0.6354**	**0.6354**	**0.6354**	**0.6354**	**0.6354**	
	D_s	**0.1516**	**0.1516**	**0.1516**	**0.1516**	**0.1516**	**0.1516**	**0.1516**	**0.1516**	
	A_s	**0.2118**	**0.2118**	**0.2118**	**0.2118**	**0.2118**	**0.2118**	**0.2118**	**0.2118**	

Table 3: Percentage of gained efficiency by the I,D,A-symmetrical (leftmost) and I_D,D_s,A_s-symmetrical designs (rightmost), with respect to the best design for each of the six criteria. The best designs are selected among the ones in Tables 1 and 2.

		I-opt	D-opt	A-opt		I_D-opt	D_s-opt	A_s-opt
Instance 1	I	-1.11	30.46	1.69	I_D	-3.00	5.49	5.49
	D	1.13	-7.35	-0.90	D_s	5.07	-1.87	-1.87
	A	0.68	17.32	-0.17	A_s	6.73	-3.18	-3.18
Instance 2	I	-0.90	20.57	-0.40	I_D	-1.32	8.84	8.84
	D	2.62	-10.57	1.36	D_s	3.80	-1.27	-1.27
	A	1.48	17.55	-0.39	A_s	4.50	-3.04	-3.04
Instance 3	I	-0.81	64.93	0.48	I_D	-1.91	11.59	11.59
	D	4.25	-11.06	1.46	D_s	9.37	-2.01	-2.01
	A	-0.27	44.51	-1.52	A_s	10.35	-2.66	-2.66
Instance 4	I	-1.18	0.05	0.05	I_D	0.00	0.14	0.00
	D	20.32	-0.08	-0.08	D_s	0.00	-0.10	0.00
	A	10.30	0.00	0.00	A_s	0.00	0.06	0.00
Instance 5	I	0.00	0.00	0.00	I_D	0.00	0.00	0.00
	D	0.00	0.00	0.00	D_s	0.00	0.00	0.00
	A	0.00	0.00	0.00	A_s	0.00	0.00	0.00

the optimization. We observe that the executions times for MS-TPLS algorithm are very good relative to the CE algorithm even when 6 criteria are being considered. We note though that updating formulas were not used in either of the algorithms.

6 Discussion

In this paper, we presented a novel algorithm, the Multi-Stratum Two-Phase Local Search (MS-TPLS), for the multi-objective optimal design of multi-stratum experiments. Our algorithm is able to concurrently optimize up to six of the most commonly used criteria, namely

Table 4: Execution times (in seconds) of the CE algorithm and of the MS-TPLS algorithm optimising 2-6 criteria.

Number of Criteria	1	2	3	4	5	6
Instance 1	931	353	716	914	1002	2067
Instance 2	2263	586	1225	1951	1781	3365
Instance 3	8410	1781	3370	5140	8000	6551
Instance 4	545	161	256	381	493	516
Instance 5	914	121	213	332	387	494

I-, D-, A-, I_D-, D_s- and A_s-optimality. The newly proposed algorithm is an extension of the algorithm we proposed in Sambo et al. (2014), improving it along two fundamental directions: the design type is extended from split-plot designs to all types of multi-stratum designs and the possible criteria to be considered for the concurrent optimization are extended from just the I- and D-criteria to the six aforementioned criteria. Another advantage of our proposed solution is that the algorithm can be easily modified in order to use different optimal criteria and, consequently, to tackle many different applications.

Our algorithm allows one to study the entire Pareto front of the optimization problem and to select the designs representing the desired trade-off between the competing objectives. Although only one solution can be implemented in practice, the Pareto approach has advantages when the decision maker's preference is not known a priori: no matter what this preference is, the solution that will be optimal under this preference is a Pareto-optimal solution, and providing the Pareto front can help eliciting the decision maker's preference by presenting them a set of trade-off solutions (Tricoire, 2012). However, when no preference is available, we pointed out what we name the *I,D,A-symmetrical and I_D,D_s,A_s-symmetrical design*, i.e. the design from the Pareto front which is the closest to the ideal point optimizing all considered in the prefix criteria, is a good representative of the entire Pareto front and a candidate design to select when no other information can be used to guide the choice.

It is important to underline, since our algorithm is based on different weighted sums of multiple objective functions into a single scalar function, that with only one weighted sumit is quite impossible to know the correct weights needed to generate points evenly spread on the Pareto curve without actually knowing the shape of the Pareto curve (Das and Dennis, 1997). To overcome this limitation, our algorithm runs a certain number of scalarizations in order to explore the search space as much as possible. Other solutions are available in the literature such as the multi-directional local search (MDLS) proposed by Tricoire (2012). A key idea of MDLS is to use different local searches, each of them working on a single objective avoiding the need for a weighted sum. As a future research direction, a careful comparison between different approaches could be done in order to find novel solutions for multi-objective optimization.

We assessed the behaviour of our algorithm on five different problem instances drawn from the literature, spanning different types of models and of multi-stratum designs. For our analyses, we choose to jointly optimize the I, D and A criteria and the I_D, D_s and A_s criteria, because of the similarities between the criteria in each of the two groups. Our algorithm would have allowed us to concurrently optimize up to six criteria, but we limited ourselves to groups of three to facilitate the representation and accessibility of the results.

Currently, the values of the variance ratios $\eta_i, i = 1, \ldots, s$ are inputs of the algorithm. However, the MS-TPLS algorithm can be easily applied using optimality criteria that can handle uncertainty about the variance ratios (see Mylona et al. (2014)). In Lu and Anderson-Cook (2014) and Lu et al. (2014), a Pareto front based algorithm is presented that can handle up to four criteria relative to the cost and the robustness to the error variance ratio for split-plot experiments.

From our analyses of the Pareto front shape in the different cases it emerged that the use of a multi-objective approach for the simultaneous optimization of different criteria is worthwhile when considering complex model structures, such as the second-order model and, to a lesser extent, a model with main effects and interactions. If the design is based on a simple model, such as the main effect model, choosing just one criterion and optimizing it seems to be enough, as the optimal designs for each criterion coincide.

References

I. Das and J.E. Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization. *Structural Optimization*, 14(1):63–69, 1997.

J. Dubois-Lacoste, M. López-Ibáñez, and T. Stützle. Improving the anytime behavior of two-phase local search. *Annals of Mathematics and Artificial Intelligence*, 61(2):125–154, 2011. doi: 10.1007/s10472-011-9235-0.

L. F. S. Escouto. Use of fermented cassava starch in the development of a formulation for bread without gluten. In 20^{th} *International Symposium of the Society for Tropical Root Crops, ISTRC 2000, Japan*, 2000.

L. Ferryanto and N. Tollefson. A split-split-plot design of experiments for foil lidding of contact lens packages. *Quality Engineering*, 22(4):317–327, 2010.

S. Gilmour and L. A. Trinca. Optimum design of experiments for statistical inference. *Journal of the Royal Statistical Society Series C-Applied Statistics*, 61(3):345–401, 2012.

S. G. Gilmour and L. A. Trinca. Some practical advice on polynomial regression analysis from blocked response surface designs. *Communications in Statistics: Theory and Methods*, 29:2157–2180, 2000.

P. Goos. *The Optimal Design of Blocked and Split-plot Experiments*. New York: Springer, 2002.

P. Goos. Optimal versus orthogonal and equivalent-estimation design of blocked and split-plot experiments. *Statistica Neerlandica*, 60:361–378, 2006.

P. Goos and A.N. Donev. The D-optimal design of blocked experiments with mixture components. *Journal of Quality Technology*, 38:319–332, 2006.

P. Goos and M. Vandebroek. Optimal split-plot designs. *Journal of Quality Technology*, 33:436–450, 2001.

P. Goos and M. Vandebroek. D-optimal split-plot designs with given numbers and sizes of whole plots. *Technometrics*, 45:235–245, 2003.

R. H. Hardin and N. J. A. Sloane. Computer-generated minimal (and larger) response-surface designs: (II) The cube. Technical report, 1991. http://www2.research.att.com/~njas/doc/meatball.pdf.

R. H. Hardin and N. J. A. Sloane. A new approach to the construction of optimal designs. *Journal of Statistical Planning and Inference*, 37:339–369, 1993.

B. Jones and P. Goos. A candidate-set-free algorithm for generating D-optimal split-plot designs. *Journal of the Royal Statistical Society Series C-Applied Statistics*, 56:347–364, 2007.

B. Jones and P. Goos. D-optimal design of split-split-plot experiments. *Biometrika*, 96:67–82, 2009.

B. Jones and P. Goos. I-optimal versus D-optimal split-plot response surface designs. *Journal of Quality Technology*, 44:85–101, 2012.

B. Jones and C. J. Nachtsheim. Split-plot designs: What, why, and how. *Journal of Quality Technology*, 41:340–361, 2009.

L. Lee Ho, C. A. Vivacqua, and A. L. Santos de Pinho. Split-plot type designs for physical prototype testing. Working Paper ni12024, University of Cambridge, Isaac Newton Institute for Mathematical Sciences, 2012.

J. D. Letsinger, R. H. Myers, and M. Lentner. Response surface methods for bi-randomization structures. *Journal of Quality Technology*, 28:381–397, 1996.

L. Lu and C. M. Anderson-Cook. Balancing multiple criteria incorporating cost using Pareto front optimization for split-plot designed experiments. *Quality and Reliability Engineering International*, 30(1):37–55, 2014.

L. Lu, C. M. Anderson-Cook, and T. J. Robinson. Optimization of designed experiments based on multiple criteria utilizing a Pareto frontier. *Technometrics*, 53(4):353–365, 2011.

L. Lu, T. J. Robinson, and C. M. Anderson-Cook. A case study to select an optimal split-plot design for a mixture-process experiment based on multiple objectives. *Quality Engineering*, 26(4):424–439, 2014.

K. Mylona, P. Goos, and B. Jones. Optimal design of blocked and split-plot experiments for fixed effects and variance component estimation. *Technometrics*, 56(2):132–144, 2014.

L. Paquete and T. Stützle. Stochastic local search algorithms for multiobjective combinatorial optimization: A review. In T. F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics*, Computer and Information Science Series, pages 29–1—29–15. Chapman & Hall/CRC, Boca Raton, FL, 2007.

F. Sambo, M. Borrotti, and K. Mylona. A coordinate-exchange two-phase local search algorithm for the d- and i-optimal design of split-plot experiments. *Computational Statistics and Data Analysis*, 71:1193–1207, 2014.

F. Tricoire. Multi-directional local search. *Computers and Operations Research*, 39(12):3089–3101, 2012.

L. A. Trinca and S. G. Gilmour. Difference variance dispersion graphs for comparing response surface designs with applications in food technology. *Journal of the Royal Statistical Society Series C-Applied Statistics*, 48(4):441–455, 1999.

L. A. Trinca and S. G. Gilmour. Multi-stratum response surface designs. *Technometrics*, 43:25–33, 2001.

L. A. Trinca and S. G. Gilmour. Improved split-plot and multi-stratum designs. *Technometrics*, 57(2):145–154, 2015.